International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Enhancing Real-Time Vision-Based Sign Language Interpretation: A Deep Learning Approach

Irfanali J. Shaikh[1*], Dr. Prasanna Shete[2]

Submitted: 03/05/2024 **Revised**: 16/06/2024 **Accepted**: 23/06/2024

Abstract: Sign language interpretation via real-time vision-based systems presents a complex challenge due to the intricate nature of sign language gestures and the variability in human motion. Effective interpretation requires robust systems that can handle the nuances of visual data and translate them into comprehensible text or speech. This study explores the efficacy of various deep learning architectures in improving the accuracy and reliability of sign language interpretation. Specifically, the application of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) alongside simpler Artificial Neural Networks (ANN) with different activation functions such as Rectified Linear Unit (ReLU) and Leaky ReLU (LReLU). Through experiments, research shows that LSTM and GRU are particularly effective for continuous frame data due to their ability to process temporal sequences, simpler ANNs with targeted hyperparameter tuning for static frames. The study provides a comparative analysis, revealing that GRU outperforms LSTM in handling short sequences, and that there is negligible performance difference between ANNs using ReLU and LReLU for single-frame interpretation. Findings contribute to the ongoing efforts to refine and enhance technological solutions for the deaf and mute communities, ensuring more accessible and effective communication tools. The research underscores the importance of clean input data and highlights specific preprocessing techniques that aid in focusing on relevant data points, thus significantly boosting the performance of vision-based sign language interpretation systems.

Keywords: *Sign Language Interpretation, Deep Learning, Gated Recurrent Unit (GRU), Real-Time Vision-Based Systems*

1. Introduction:

In the realm of communication, sign language stands as a vital conduit for the deaf and hard-of-hearing communities. Despite its importance, persistent barriers in automated sign language interpretation have limited its accessibility and integration into digital communication platforms. Recent advances in deep learning and computer vision have opened new avenues for addressing these challenges, providing the potential to revolutionize how sign language is interpreted in real-time. This study explores the development of a deep learning-based system designed to enhance the accuracy and speed of vision-based sign language interpretation. By leveraging state-of-the-art neural network architectures, recurrent neural networks (RNNs), this system aims to decode complex sign language gestures from video input with high precision. The integration of these technologies addresses critical issues such as varying lighting conditions, diverse signer backgrounds, and a wide range of signing speeds. Furthermore, this approach underscores the importance of scalable and adaptable models that can learn from a vast dataset of sign language gestures, ensuring inclusivity and robustness.

Through a detailed examination of model architecture, training processes, and real-world application scenarios, this study aims to contribute significantly to the field of accessible communication technologies, bridging the gap between technological advances and community needs. Real-time vision-based sign language interpretation using deep learning has emerged as a pivotal technology to bridge communication gaps between the deaf community and the hearing majority. This approach leverages advanced deep learning techniques to accurately and efficiently translate sign language into text or speech, significantly enhancing accessibility and inclusivity. Traditional methods of sign language interpretation, which rely heavily on human interpreters or sensor-based systems, face limitations such as high costs, limited availability, and user-unfriendliness (Aloysius & Geetha, 2020).

Recent advancements in deep learning and computer vision have enabled the development of robust models capable of interpreting sign language with high accuracy. Recurrent neural networks (RNNs) have been particularly effective in recognizing and translating sign language gestures. For instance, the transformer model combined with ResNet50 embeddings has outperformed traditional sequence-to-sequence models in translating German, American, and Chinese sign languages (Ananthanarayana et al., 2021). Moreover, integrating sensor fusion

[1*]*Research Scholar1, Department of Computer Engineering, KJSCE, Vidyavihar, Mumbai-400077, India, Email: irfanali.s@somaiya.edu*
Professor2, Department of Computer Engineering, KJSCE, Vidyavihar, Mumbai-400077, India
[2]*prasannashete@somaiya.edu*
***Corresponding Author:** Irfanali J. Shaikh*
**Research Scholar1, Department of Computer Engineering, KJSCE, Vidyavihar, Mumbai-400077, India*
Email: irfanali.s@somaiya.edu

techniques with deep learning has proven to be a promising approach. By combining inertial measurement units (IMUs), researchers have achieved impressive recognition rates for dynamic ASL gestures, demonstrating the potential for real-time applications without the constraints of visual angles (Lee, Chong, & Chung, 2020). The deployment of these deep learning models in real-time environments has been facilitated by tools such as the MediaPipe library and LSTM algorithms, which have shown substantial accuracy in recognizing motion-based ASL phrases and can be integrated into mobile applications for practical use (Ru & Sebastian, 2023). In addition to the advancements in deep learning models for static and dynamic sign language recognition, research has shown the benefits of employing hybrid approaches. These approaches combine multiple deep learning techniques to improve overall system performance. For instance, hybrid models utilizing both Long Short-Term Memory (LSTM) networks for capturing temporal dynamics for spatial feature extraction have demonstrated remarkable accuracy and robustness in recognizing continuous sign language sequences (Cui, Liu, & Zhang, 2017).

Sensor-based methods integrated with deep learning have addressed several challenges associated with vision-based systems. By employing inertial measurement units (IMUs), researchers have achieved high recognition rates for dynamic ASL gestures, thus offering solutions that are not constrained by visual angles and environmental factors (Lee, Chong, & Chung, 2020). This sensor fusion approach demonstrates the potential for developing smart wearable systems that can provide real-time sign language interpretation in diverse settings. The development and deployment of real-time sign language interpretation systems have been facilitated by advanced software libraries and frameworks. For instance, the use of the MediaPipe library combined with LSTM algorithms has enabled the creation of mobile applications capable of recognizing motion-based ASL phrases with substantial accuracy (Ru & Sebastian, 2023). These applications highlight the feasibility of integrating deep learning models into user-friendly platforms for practical use. Another notable advancement in the field is the use of transfer learning and vision transformers. These techniques have been particularly effective in recognizing complex sign languages, such as Arabic Sign Language (ArSL). Transfer learning models, such as ResNet and InceptionResNet, along with vision transformers like ViT and Swin, have demonstrated high accuracy and efficiency in classifying sign language gestures, thus showcasing the potential for applying these methods to low-resourced languages (Alharthi & Alzahrani, 2023).

A comprehensive review of significant research on enhancing real-time vision-based sign language interpretation using deep learning approaches. The studies cover various techniques, including sensor fusion, recurrent convolutional neural networks, and transfer learning. These studies collectively contribute to the advancement of accurate and efficient sign language recognition systems as shown in table1.

Table 1: Summary of Research studies on Deep Learning Approaches for Sign Language Interpretation

Title	Authors	Year	Summary
Understanding Vision-Based Sign Language Recognition	Aloysius & Geetha	2020	Examines traditional and vision-based methods for sign language recognition, highlighting limitations and potential.
Deep Learning Methods for Sign Language Translation	Ananthanarayana et al.	2021	Compares various deep learning models like CNNs and sequence-to-sequence models for sign language translation.
Sensor Fusion of Motion-Based Sign Language Interpretation	Lee, Chong, & Chung	2020	Utilizes sensor fusion combining IMUs with CNNs to recognize dynamic ASL gestures, achieving high accuracy.
Real-Time American Sign Language (ASL) Interpretation	Ru & Sebastian	2023	Employs MediaPipe and LSTM algorithms to create a mobile app for recognizing motion-based ASL phrases.
Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition	Cui, Liu, & Zhang	2017	Uses RCNNs for continuous sign language recognition, translating sign sequences into text or speech.
Vision Transformers and Transfer Learning Approaches for Arabic Sign Language	Alharthi & Alzahrani	2023	Applies transfer learning and vision transformers to recognize Arabic Sign Language gestures, demonstrating high efficiency.

The studies summarized in this table1 shows the significant advancements made in the field of vision-based sign language interpretation through deep learning approaches. The integration of various techniques, such as sensor fusion, recurrent convolutional neural networks, and transfer learning, has demonstrated substantial improvements in the accuracy and efficiency of sign language recognition systems.

Indian Sign Language (ISL) interpretation is a crucial means of communication for the deaf and mute community in India. It involves the use of visual-manual modality to convey meaning, as opposed to the auditory-verbal communication used by hearing individuals. ISL interpretation enables people who are deaf or hard of hearing to access information, education, and engage in everyday communication, thereby playing a vital role in their social inclusion and equal participation in society. Sign language interpretation remains a vital and evolving field in the realm of communication technologies, particularly for the hearing and vocally impaired community. Recent advancements have focused on leveraging spatial-temporal models and neural networks to enhance the accuracy and effectiveness of Indian Sign Language (ISL) interpretation. The integration of these advanced computational models promises to bridge communication gaps and offer new avenues for interaction and understanding.

Research on a Deep Convolution Neural Network Model for ISL Classification highlights the significant progress in solving image classification problems. This model, evaluated on a large dataset of sign images, demonstrates high accuracy, showcasing the model's efficiency in recognizing various static signs (Dangarwala & Hiran, 2020). Wadhawan and Kumar (2020) focused on the recognition of static signs in ISL using deep learning-based convolutional neural networks. Launching public awareness campaigns to dispel myths and stereotypes about deafness and sign language, highlighting the capabilities and achievements of the deaf community. Fostering partnerships between governmental bodies, non-governmental organizations (NGOs), educational institutions, and the private sector to support and promote ISL and deaf culture. Mittal et al. (2019). This research proposes a modified LSTM model for continuous sign language recognition, focusing on sequences of connected gestures in Indian Sign Language.

The journey towards fully realizing the potential of Indian Sign Language interpretation is ongoing. It requires the collective effort of the government, private sector, educational institutions, and the community.

2.Literature Review

The relationships between different sign languages, suggesting that, despite lateral transmission and interference, some sign languages, like ASL and French Sign Language (FSL), descend from common ancestors. This concept is essential for tracing the historical roots of ISL and its connections to other sign languages (Reagan, 2021). Sign languages naturally emerge wherever there are deaf people. The history of ASL, for example, sheds light on how sign languages develop and gain recognition over time. This context is crucial for understanding the evolution of ISL and its place within the global sign language family (Fischer, 2015). Historical linguistics offers insights into the development of sign languages, with research often focusing on how these languages have evolved over time. Although the direct history of ISL might not be extensively documented, the study of sign languages from a historical perspective is crucial for understanding the evolution of ISL (Ruben, 2005).

2.1 Early methods of ISL communication and their evolution.

The theory that language evolved from manual gestures, with evidence from signed languages sharing essential linguistic characteristics with spoken languages, supports the notion of sign languages' deep-rooted history. This perspective is essential for understanding the foundational aspects of ISL's development (Corballis, 2008). Advances in technology have enabled the development of sign language recognition systems, aiming to bridge the communication gap for deaf and mute individuals. Early methods focused on recognizing binary positions of fingers to convert signs into text, demonstrating the evolving nature of ISL communication methodologies (Rajam & Balakrishnan, 2011). The evolution of the manual alphabet, tracing back to the monks of the seventh century, played a significant role in the development of sign languages, including ISL. The adaptation of these systems over centuries underscores the historical depth of sign language communication (Padden & Gunsauls, 2003). Literature reviews on ISL recognition systems highlight the field's growth, reflecting on the natural evolution of sign languages within communities and the complexity of developing recognition systems for ISL compared to other sign languages. This body of work emphasizes the uniqueness and challenges of ISL's evolution (Dour & Sharma, 2015).

2.2 Technological Advancements in ISL Interpretation

Recent advancements have significantly improved the efficacy and accessibility of ISL interpretation. Peguda et al. (2022) developed a model that converts speech into ISL for six regional Indian languages, addressing the communication barriers faced by the hearing-impaired and mute individuals by displaying corresponding gestures as outputs. This model represents a significant step towards making spoken languages accessible to the

deaf community. Dutta et al. (2015) introduced a system that captures double-handed ISL as a series of images, processing them to generate speech and text. This innovation provides a voice for the speechless, further bridging the gap between the deaf-mute community and the hearing world. Goyal et al. (2013) proposed a sign language recognition system employing the SIFT algorithm for feature extraction from real-time images, achieving a 95% accuracy rate for certain alphabets. This system demonstrates the potential of machine learning in enhancing sign language recognition accuracy. Dias et al. (2022) developed SignEnd, an ISL system that recognizes alpha-numeric hand signs of users with both five and six fingers, translating them into text equivalents. The system showcases the use of custom datasets and advanced machine learning models to accommodate the diverse needs of the deaf-mute community with an average accuracy of 90%.

2.3 Milestones in the development of ISL interpretation technologies.

Initial research efforts in ISL recognition highlighted the complexity of sign language, distinguishing between single and double-handed signs and the importance of developing systems that could accurately interpret these signs. Early systems relied on methods like Artificial Neural Networks (ANN), Support Vector Machine (SVM), and Hidden Markov Models (HMM) for sign recognition (Nair & Bindu, 2013). The development of a novel vision-based gesture recognition system marked a significant advancement, offering a signer-independent model capable of recognizing both static and dynamic gestures from live video feeds. This technology demonstrated improved accuracy rates for recognizing finger spelling alphabets and single-handed dynamic words, signifying a step forward in real-time ISL interpretation (Athira, Sruthi, & Lijiya, 2019). Efforts to develop real-time ISL recognition systems have been crucial in making ISL interpretation more accessible and immediate for the deaf and mute communities. These systems utilize skin segmentation and machine learning algorithms to recognize signs from live video, allowing for more natural and fluid communication.

Integrating neural networks with optimization algorithms like Genetic Algorithm (GA), Evolutionary Algorithm (EA), and Particle Swarm Optimization (PSO) has significantly improved the accuracy of ISL gesture recognition. This approach has resulted in systems that not only recognize ISL gestures more effectively but also do so with a high degree of precision, demonstrating the potential for further advancements in the field (Hore et al., 2015). The application of deep learning-based convolutional neural networks (CNN) to recognize ISL gestures, especially for static signs, marks a leap forward in the accuracy and efficiency of interpretation systems. By collecting a vast dataset of sign images and evaluating the system across numerous CNN models, researchers have achieved unprecedented levels of accuracy, showcasing the effectiveness of deep learning in ISL interpretation (Sharma & Singh, 2021). Advances in wearable technology, integrating sensor fusion for sign language interpretation, represent an innovative approach to ISL recognition.

The development of gesture recognition algorithms for translating ISL into English represents a milestone in making ISL more comprehensible to the non-signing public. By utilizing a combination of data acquisition, pre-processing, and template matching techniques, these systems translate ISL gestures into English text or speech, facilitating easier communication between deaf individuals and those unfamiliar with sign language. Such technologies play a crucial role in breaking down communication barriers and fostering more inclusive interactions. Vision-based recognition systems have emerged as a powerful tool for interpreting ISL, enabling efficient human-computer interaction and helping bridge the communication gap between hearing-impaired individuals and the broader society. Such systems utilize hand tracking, segmentation, feature extraction, and classification techniques to interpret hand gestures accurately (Ghotkar & Kharate, 2014). Machine learning algorithms, including Artificial Neural Networks (ANN) and Support Vector Machine (SVM) classifiers, have been extensively applied to ISL recognition, achieving remarkable accuracy rates. These developments underscore the potential of machine learning in enhancing the efficiency and reliability of ISL interpretation systems (Ekbote & Joshi, 2017). The integration of Internet of Things (IoT) technology with ISL interpretation systems has opened new avenues for communication aids for the deaf and mute. By leveraging IoT, these systems offer enhanced connectivity and accessibility, enabling more effective communication solutions for individuals with hearing and speech impairments.

Leveraging image processing and machine learning techniques has proven effective in creating reliable communication interpretation programs for ISL. These technologies enable the conversion of sign language gestures into readable outputs, bridging the communication gap between deaf-mute individuals and those unfamiliar with sign language. The use of image processing for gesture recognition, combined with machine learning for gesture classification, highlights the interdisciplinary approach to enhancing ISL interpretation (Apoorv, S. et al., 2020). Developing systems that can translate speech to ISL represents a significant leap towards making information and services more accessible to the hearing-impaired. These translation systems aim to eliminate the need for written

texts as the sole mode of communication, thereby facilitating an educational tool for learning ISL and improving overall access to information. This advancement is particularly crucial in bridging the communication divide, offering a platform for anyone to communicate without prior knowledge of ISL (Kulkarni et al., 2021). The application of Artificial Neural Networks (ANN) for the recognition of Indian sign language emphasizes the role of advanced computational techniques in enhancing the accuracy of sign language interpretation. By focusing on fingerspelling and word-level signs, ANN-based methods provide a robust framework for automatically recognizing and interpreting ISL, thereby facilitating smoother communication for the deaf-dumb community and reducing their dependence on interpreters (Adithya et al., 2013).

3. Methodology

This study concentrates on translating Indian Sign Language (ISL) into text and spoken language. It employs advanced deep learning models, including LSTM, GRU, RNN. Specifically, the GRU and ANN models are selected for training using a ReLu optimizer. A specialized dataset has been developed from video recordings for this purpose as given link https://github.com/Irfanali-shaikh/ISL/tree/main

Despite the promising advances in the field, the interpretation of sign language involves several significant challenges that must be meticulously addressed to guarantee the accuracy and reliability of the communication systems developed. These challenges encompass issues such as inadequate lighting conditions, limited availability of comprehensive datasets, the necessity for continuous frame analysis, network latency, and interference from noisy backgrounds. Each of these factors can adversely affect the performance of sign language recognition systems, thereby complicating the interaction between deaf-mute individuals and the general populace.

This study addresses the persistent communication barriers that individuals with hearing and speech impairments face, particularly through the lens of Indian Sign Language. By delving into a variety of innovative solutions, this research aims to contribute significantly to the development of more accessible and dependable methods of communication. The study provides a thorough examination of both the prevailing state of research and the challenges encountered in the field of sign language interpretation. Central hypothesis posits that by utilizing sophisticated deep learning algorithms coupled with precise data preprocessing techniques, it is possible to markedly enhance the accuracy of sign language interpretation. This enhancement will not only benefit the deaf and mute community by making communication tools more accessible but also facilitate smoother interaction with the general population. Motivated by the critical need to bridge the communication gap for individuals with hearing and speech impairments, this research contributes to the academic and practical fields by presenting cutting-edge solutions, tackling prevalent challenges, and advancing a hypothesis with the potential to revolutionize sign language interpretation. Through these efforts, research aim to foster a more inclusive and effective communicative environment for all individuals, particularly those within the deaf and mute communities.

3.1 3D Sign Language Detection

Utilizing the advancements in 3D data acquisition, this approach employs a 9-camera motion capture system to capture comprehensive sign language gestures. Traditional deep learning architectures such as Hidden Markov Models (HMM), Long Short-Term Memory (LSTM), and have been adapted to process this 3D data effectively. This system employs angular velocity between joints, utilizing Joint Angular Velocity Maps (JAVM) to enhance recognition accuracy, which has achieved an impressive average accuracy of 95.65%.

3.2 A-SLR using SVM

This method focuses on recognizing specific characters from American Sign Language (ASL) amidst challenging environments and noisy backgrounds. By employing HSV skin color segmentation and Principal Component Analysis (PCA), this approach uses Support Vector Machines (SVM) for the recognition process. The system achieves a remarkable accuracy rate of 99.4% for characters B, D, F, L, and U, with significant improvements facilitated by a self-prepared dataset and sophisticated noise removal algorithms.

3.3 C-SLR Based on Video Sequence

This approach is tailored for recognizing Chinese hand gestures, leveraging Bidirectional Long Short-Term Memory (BLSTM) and residual networks. The methodology involves segmenting the region of interest, extracting spatiotemporal features, and classifying video sequences. The B3D ResNet technique, which combines BLSTM and 3D residual networks, plays a crucial role in interpreting sign language, achieving an average accuracy of 88.35% on two datasets.

3.4 Modified LSTM for Continuous SLR Using Leap Motion

Utilizing Leap Motion sensors for capturing 3D hand motion, this method employs a modified 4 Gated LSTM model to process isolated sign words and sentences. The data is trained using CONV2D, and the modified model demonstrates superior performance compared to

traditional LSTM, with accuracy rates of 89.50% for sign words and 72.30% for sign sentences.

3.5 C-SL Alphabet Based on Random Forest

This unique approach utilizes surface Electromyography (sEMG) data from the forearms to recognize muscle motion related to alphabet characters. Employing the Random Forest algorithm, this method achieves a notable accuracy rate of nearly 95.48%, surpassing the performance of other algorithms such as Support Vector Machine (SVM) and Artificial Neural Network (ANN).

3.6 LSTM-HMMs to Discover Sequential Parallelism in SLV

Focusing on the dual attributes of hand and lip movements in sign language, this methodology combines Long Short-Term Memory (LSTM) and Hidden Markov Models (HMMs). Utilizing the RWTH-PHOENIX weather 2014 dataset, the approach achieves an average accuracy of 73%, demonstrating its potential in recognizing sequential parallelism in sign language videos.

3.7 Vision-Based Continuous Sign Language Recognition

Aiming to enhance accuracy in vision-based sign language recognition, this research explores various methodologies, including graph-based models and Dynamic Time Warping (DTW). The study also addresses the challenges associated with noisy backgrounds, motion, relocations, and lighting issues, emphasizing the importance of data size in achieving reliable recognition results.

3.8 Virtual Sign Channel for Deaf and Mute Users

This innovative approach facilitates sign-to-text and textual-to-sign language conversion, accommodating users of different European sign languages. The integration of gloves with gyroscopic sensors and Microsoft Kinect for skeleton data extraction has yielded a remarkable accuracy rate of 86%, showcasing the potential of virtual channels in bridging communication gaps. The architectural framework and operational workflow of proposed sign language interpretation system, designed to transform sign language into comprehensible text or speech.

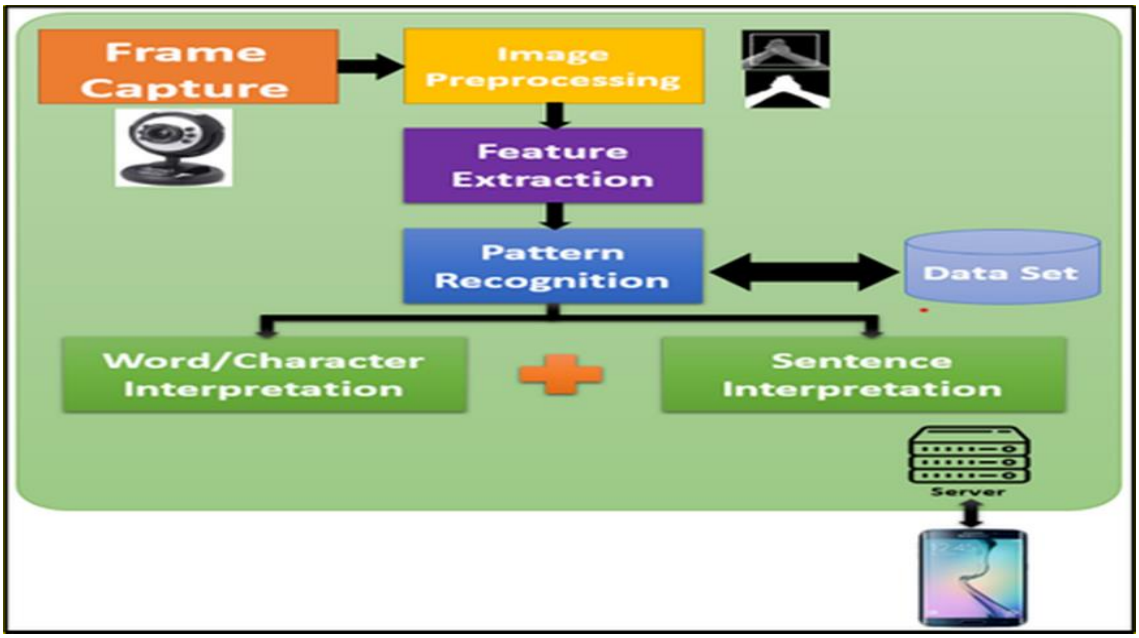


Fig 1. Architecture of Proposed Method

The architecture, depicted in Figure 1, elaborates on each step involved in the process, from frame acquisition to the final interpretation.

I. Frame Acquisition

The process begins with the acquisition of frames or sets of frames captured by a camera module connected to the local system. This stage is crucial as it forms the foundation for all subsequent processing stages by providing the raw visual data necessary for sign language recognition.

II. Pre-processing

Once the frames are captured, they undergo a series of pre-processing steps. This includes the elimination of background noise and the extraction of critical facial and hand points from the frames. The extraction is facilitated by advanced tools such as mediapipe, which are instrumental in isolating and highlighting the features essential for accurate sign language interpretation.

III. Feature Extraction

The third step involves the extraction of features from the pre-processed frames. This includes the normalization of pixel values within the frames to standardize the data input into the machine learning models. Normalization is key to ensuring that the system can effectively interpret the sign language gestures irrespective of varying lighting conditions and other environmental factors.

IV. Data Set and Pattern Recognition

This step is divided into two interconnected components: the Dataset and Pattern Recognition. During the model training phase, a set of sample data, which could be static images or video sequences, is used to teach the system how to recognize patterns associated with different signs. This phase is critical for developing the system's ability to autonomously recognize and interpret sign language from new input.

V. Interpretation

The final step in the system's workflow involves interpreting the patterns recognized in the previous phase. This interpretation could be at the level of characters, words, or sentences, depending on the input

frame's complexity. For example, in a phrase like "Nafis is a good boy," character interpretation might be used for recognizing individual letters in "Nafis," while word interpretation could be applied to "good" and "boy." Moreover, sentence framing, which involves more complex Natural Language Processing (NLP) techniques, could be utilized to construct meaningful sentences, although NLP is considered beyond the current scope of this study.

System Deployment

The entire system is designed to operate on a server, allowing for the processing of frames captured through any mobile computing device. This server-based processing enables the system to return interpreted sign language in the form of text or speech to the user. Focus is on implementing this system on local devices, providing a robust and accessible platform for real-time sign language interpretation.

3.9 Models and Data Collection

For practical implementation, developed two distinct models: one based on a set of frames (video) for word recognition and another based on a single frame for character recognition. Different datasets have been collected to tailor the training process to specific needs and requirements of sign language interpretation.

This structured approach not only highlights the evolution of sign language recognition techniques but also emphasizes the importance of efficient feature extraction, noise reduction, and the use of advanced technology to enhance the accuracy and accessibility of sign language interpretation systems.

A. Dataset for Sign Language Word

When this project commenced, encountered difficulties in sourcing a suitable video dataset for sign language recognition. Available datasets often exhibited inconsistencies and variable frame lengths, which compromised their utility for specific requirements. To address this issue, we opted to construct own dataset, prioritizing control and consistency. Stored the data in .npy format to minimize storage size. Below is a description of the dataset configuration:

The dataset is organized into folders corresponding to nine specific words: ['-', 'Hello', 'Good', 'Afternoon', 'Sad', 'Marriage', 'Home', 'Blind', 'Thanks']. Each word folder in the training set contains 40 sub-folders labeled from 0-39. Each sub-folder includes 30 .npy files representing the facial, pose, left hand, and right-hand data points, effectively serving as frames per video. The individual file shape is (1662,), resulting in a total shape per video of (30,1662). For each word, the collective dataset shape is (40,30,1662).

Similarly, for the testing set, each word folder contains 10 sub-folders labeled from 0-9, with each folder containing 30 .npy files. The shape for each testing video is also (30,1662), and the total shape for each word in the testing set is (10,30,1662).

Table 2: Shape of Word Datasets

Data Set	**Shape of Each video**	**shape of all videos**
Training Set	(40,30,1662)	(360,30,1662)
Testing Set	(10,30,1662)	(90,30,1662)

The breakdown of the data structure used in our sign language interpretation study as shown in table 2. For the training set, each video comprises 30 frames with 1662 data points each, arranged into 40 series, resulting in a

total shape of (360,30,1662) for all videos. The testing set follows a similar structure but with 10 series, culminating in a total shape of (90,30,1662) for all videos. This organization is critical for training and evaluating our models with consistent and structured input data as shown in Figure 2.

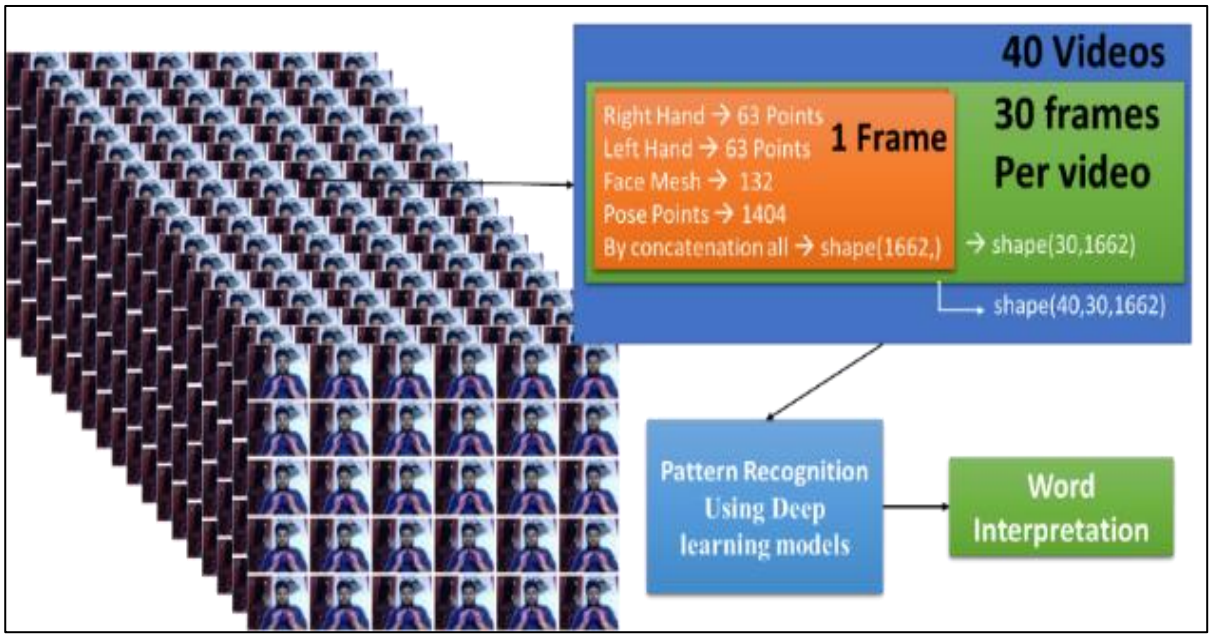


Fig 2. Shape of Word Datasets

B. Dataset for Sign Language Character

Encountering similar challenges with the video dataset for characters, also decided to create a dedicated dataset for sign language characters, focusing on data points from the left and right hands. Stored this data in .npy files to ensure efficient storage. The dataset is segmented into 28 folders, including each alphabet character, 'Space', and 'Reset'. For training purposes, collected 800 frames for each character, with each frame having a shape of (126,). Collectively, each character in the training set has a shape of (800,126). For the testing data, gathered 200 frames per character, maintaining the same individual frame shape.

Table 3: Shape of Character Datasets

Data Set	Shape of Each Character	Shape of All Characters
Training Set	(800,126)	(22400,126)
Testing Set	(200,126)	(5600,126)

The structure of the datasets used for character-level sign language recognition as shown in table 3. The training set includes data for each character structured in 800 frames, each with 126 data points, resulting in a collective shape of (22400,126) across all characters. The testing set is similarly structured but consists of 200 frames per character, aggregating to a total shape of (5600,126). This setup facilitates the detailed analysis and training of our neural network models as shown in Figure 3.

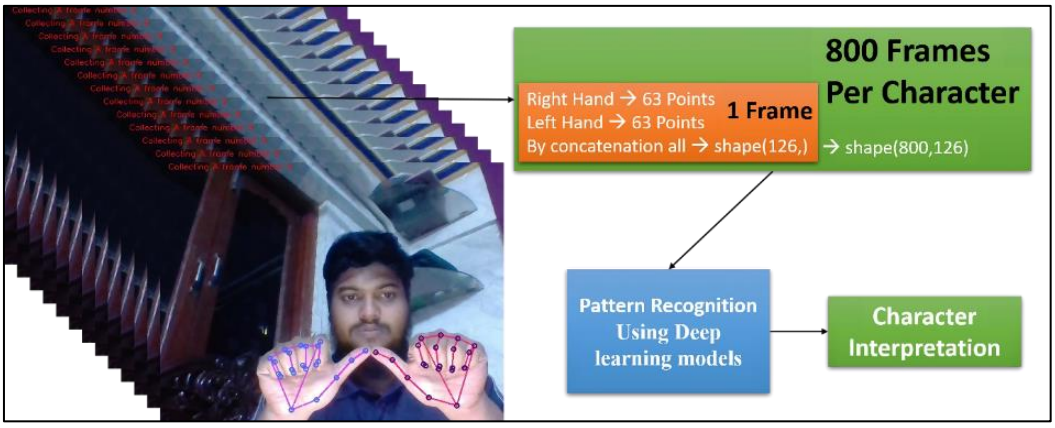


Fig 3. Shape of Character Datasets

C. Model Training

Training has been conducted on two models as previously discussed: one for word level (9 words) and another for character level (26 characters plus "Reset" and "Space"). The word-level model utilizes a set of frames and is trained using a Recurrent Neural Network, employing both Long Short-Term Memory (LSTM) and Gated Recurrent Neural Network (GRU) techniques. For the character-level recognition, a dense neural network is used, employing different activation functions (ReLU vs. Leaky ReLU) to optimize performance. This two-tiered approach allows for tailored handling of different aspects of sign language recognition, catering to the distinct demands of word and character interpretation.

4. Result and Discussion

This provides a comparative analysis of two prominent Recurrent Neural Network architectures used in study for word-level sign language recognition: Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU).

A. Long Short-Term Memory Vs Gated Recurrent Unit – Word Level

Long Short-Term Memory (LSTM):

LSTM networks are a type of Recurrent Neural Network (RNN) designed to address the vanishing gradient problem associated with standard RNNs. This capability makes them particularly suited for modeling time-series data where long-term dependencies are crucial. In study, the LSTM model incorporated multiple layers,

including dropout layers to manage outliers effectively as shown in below table 1. Below are some specific attributes of the LSTM model used:

Advantages of LSTM:

Minimization of the vanishing gradient problem due to its gated architecture. Presence of three gates (input, output, and forget gates), enhancing the model's ability to regulate information flow.

Disadvantages of LSTM:

High computational burden due to a large number of training parameters.

Greater memory requirement and slower operation compared to GRU.

Gated Recurrent Unit (GRU):

GRU, like LSTM, is designed to help capture dependencies in sequence data more effectively than standard RNNs. However, it simplifies the model architecture by using two gates (update and reset gates), which allows for faster training times and reduced memory usage. Here are the GRU model specifics:

Advantages of GRU:

Fewer training parameters, making the network more efficient in terms of memory and speed.

Generally, GRUs are faster and use less memory compared to LSTMs due to their simpler structure.

Disadvantages of GRU:

While GRUs are effective at addressing the vanishing gradient problem, they may not perform as well in tasks where long-term dependencies are more significant due to the reduced number of gates.

Model Summaries and Training Parameters:

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 30, 256)	1,965,056
dropout (Dropout)	(None, 30, 256)	0
lstm_1 (LSTM)	(None, 30, 512)	1,574,912

dropout_1 (Dropout)	(None, 30, 512)	0
lstm_2 (LSTM)	(None, 256)	787,456
dropout_2 (Dropout)	(None, 256)	0
dense (Dense)	(None, 256)	65,792
dropout_3 (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 128)	32,896
dense_2 (Dense)	(None, 9)	1,161

Total params: 4,427,273

Trainable params: 4,427,273

non-trainable params: 0

This table4 provides a detailed view of the LSTM model architecture, illustrating the layers and configurations used in training.

Model: “sequential_1”

Layer (type)	Output Shape	Param #
gru_3 (GRU)	(None, 30, 256)	1,474,560
dropout_4 (Dropout)	(None, 30, 256)	0
gru_4 (GRU)	(None, 30, 512)	1,182,720
dropout_5 (Dropout)	(None, 30, 512)	0
gru_5 (GRU)	(None, 256)	591,360
dropout_6 (Dropout)	(None, 256)	0
dense_3 (Dense)	(None, 256)	65,792
dropout_7 (Dropout)	(None, 256)	0
dense_4 (Dense)	(None, 128)	32,896
dense_5 (Dense)	(None, 9)	1,161

Total params: 3,348,489

Trainable params: 3,348,489

Non-trainable params: 0

Model: "sequential_1"

Layer (type)	Output Shape	Param #
gru_3 (GRU)	(None, 30, 256)	1,474,560
dropout_4 (Dropout)	(None, 30, 256)	0
gru_4 (GRU)	(None, 30, 512)	1,182,720
dropout_5 (Dropout)	(None, 30, 512)	0
gru_5 (GRU)	(None, 256)	591,360
dropout_6 (Dropout)	(None, 256)	0
dense_3 (Dense)	(None, 256)	65,792
dropout_7 (Dropout)	(None, 256)	0

dense_4 (Dense)	(None, 128)	32,896
dense_5 (Dense)	(None, 9)	1,161

Total params: 3,348,489

Trainable params: 3,348,489

non-trainable params: 0

Fig 6. Summary of GRU Model

This figure shows the GRU model layout, highlighting its simpler architecture relative to LSTM.

Table 4: Training Size of Algorithms for Word

Algorithm	Total Parameters for Training
LSTM	4,427,273
GRU	3,348,489

The total parameters involved in training each model, emphasizing the more lightweight nature of GRU in comparison to LSTM as shown in table4.

Through these analyses, it is evident that the choice between LSTM and GRU for a specific application should consider the trade-offs between computational efficiency and the ability to handle long-term dependencies. This discussion not only highlights the technical distinctions between the two models but also aligns their theoretical advantages and disadvantages with practical outcomes observed during their application in sign language recognition.

B. Artificial Neural Network (ReLU Vs LReLU) – Character Level

Artificial Neural Networks (ANNs) are computational models inspired by the biological neural networks found in animal brains. In this part of study, the activation functions within these networks, specifically comparing the Rectified Linear Unit (ReLU) and Leaky Rectified Linear Unit (LReLU). ReLU is defined as f(x)=max(0,x), which effectively sets all negative values to zero, allowing the network to learn faster and more effectively by introducing non-linearity. LReLU modifies the ReLU function by allowing a small, non-zero gradient when the unit is not active and x is less than zero. f(x)=max(0.01x,x). This slight slope for negative values helps to keep the gradient flow alive during the training process, which can prevent the neurons from dying out. The models using these activation functions were designed with identical parameters to isolate the effect of the activation function in experimental analysis.

Model: "sequential_5"

Layer (type)	Output Shape	Param #
dense_7 (Dense)	(None, 128)	16256
dropout_5 (Dropout)	(None, 128)	0
dense_8 (Dense)	(None, 128)	16512
dropout_6 (Dropout)	(None, 128)	0
dense_9 (Dense)	(None, 64)	8256
dropout_7 (Dropout)	(None, 64)	0
dense_10 (Dense)	(None, 32)	2080
dropout_8 (Dropout)	(None, 32)	0
dense_11 (Dense)	(None, 28)	924

Total params: 44,028

Trainable params: 44,028

non-trainable params: 0

Fig7. Summary of Designed Model for ReLu and LReLU

This figure provides a visual representation of the ANN model architecture, highlighting the configuration of layers and the placement of ReLU and LReLU functions.

C. Model Training Accuracy and Loss - Word Level

In experiments, the training accuracy and loss metrics were used to gauge the performance of the models. Accuracy measures how well the model performs, while the loss value indicates the model's error rate after each optimization iteration. The results for LSTM and GRU models trained over 10 epochs are documented in below Figure 4(a), 4(b) and Figure 5(a), 5(b).

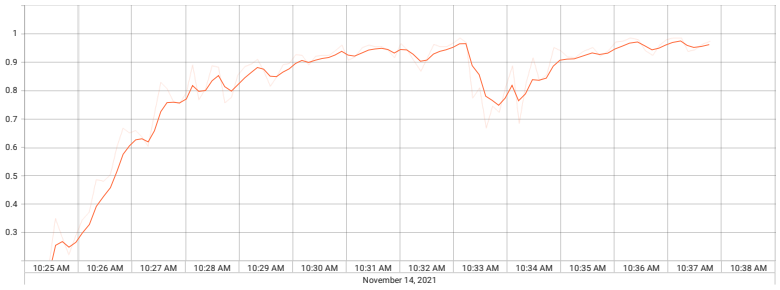


Fig 4(a). Accuracy of LSTM

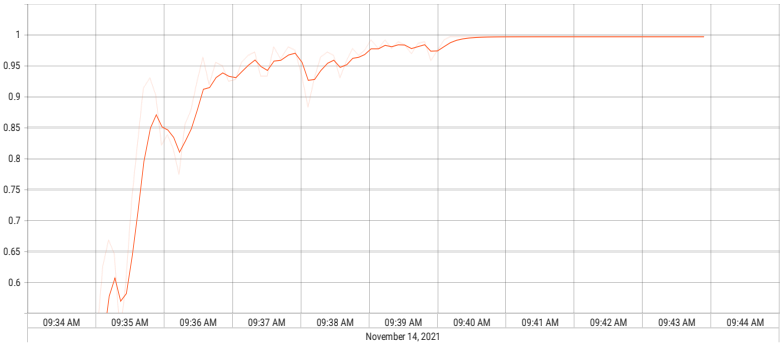


Fig 4(b). Accuracy of GRU

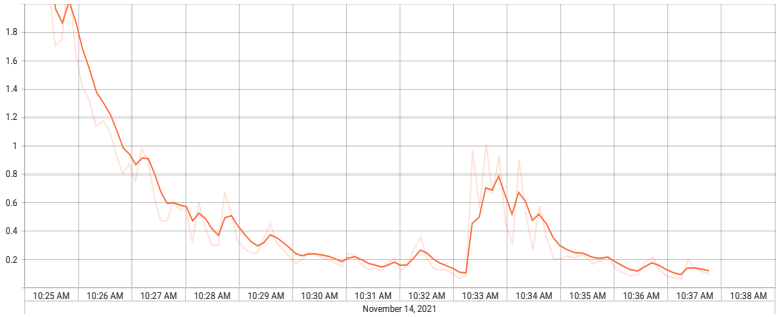


Fig 5(a). Loss of LSTM

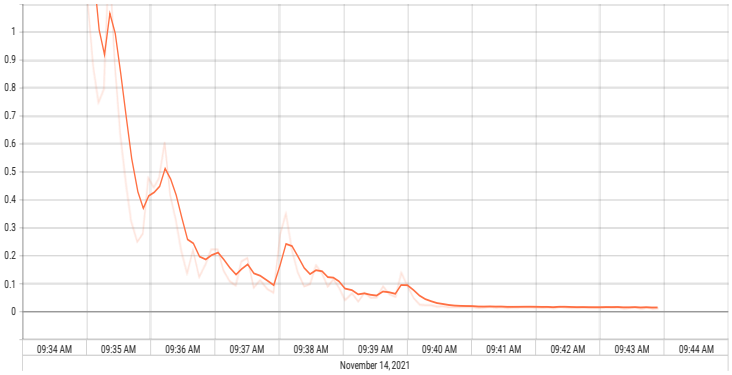


Fig 5(b). Loss for GRU

Table 5: Training Accuracy

Algorithm	Accuracy
LSTM	0.8666
GRU	0.9777

The accuracy table 5 highlights the superior performance of GRU in this setup, potentially due to its efficiency and lower complexity compared to LSTM.

Results Metrics and Visualization

In addition to accuracy and loss, calculated F1, precision, and recall scores to provide a comprehensive evaluation of the model performance. These metrics are crucial for understanding the balance between precision and recall, and the F1 score provides a harmonic mean of the two as shown in Figure 6(a) and 6(b).

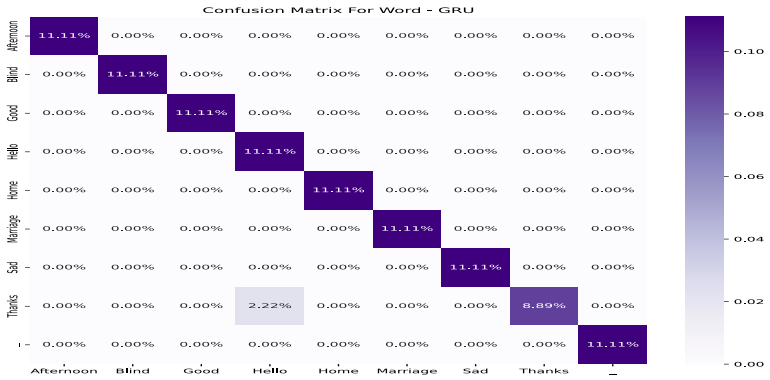


Fig 6(a): Confusion Matrix using GRU

This visualization provides insights into the true positives, true negatives, false positives, and false negatives classified by the GRU model.

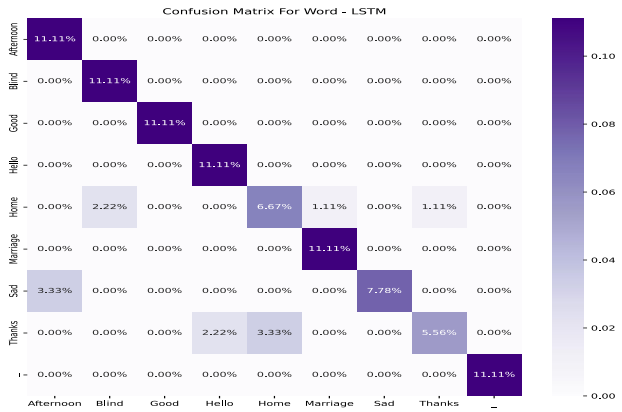


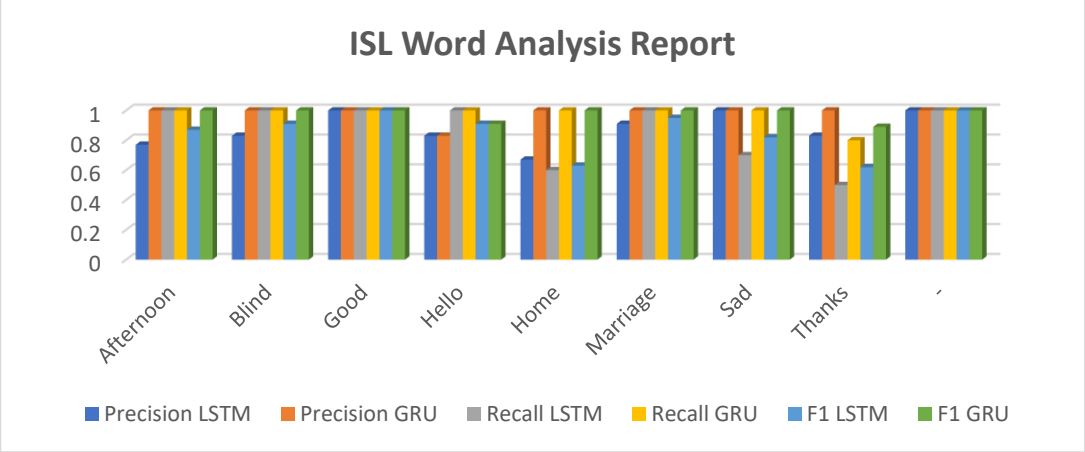
Fig 6(b). Confusion Matrix using LSTM

Table 6: Score Analysis Table - Word

Scores	Precision		Recall		F1	
Words	LSTM	GRU	LSTM	GRU	LSTM	GRU
Afternoon	0.77	1	1	1	0.87	1
Blind	0.83	1	1	1	0.91	1
Good	1	1	1	1	1	1

Hello	0.83	0.83	1	1	0.91	0.91
Home	0.67	1	0.6	1	0.63	1
Marriage	0.91	1	1	1	0.95	1
Sad	1	1	0.7	1	0.82	1
Thanks	0.83	1	0.5	0.8	0.62	0.89
-	1	1	1	1	1	1

This table 6 typically detail the precision, recall, and F1 scores for both LSTM and GRU, providing a numeric representation of the models' classification accuracy.



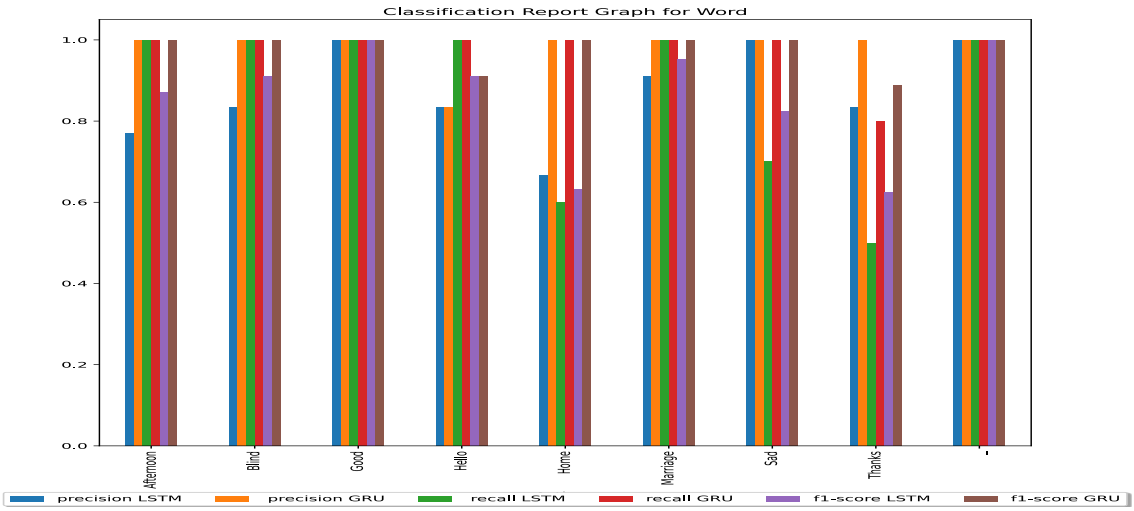


Fig 7. Score Analysis Graph - Word

This graph visualizes the comparative score analysis, offering a clear depiction of how each model performs in terms of precision, recall, and F1 score across different classes or labels as shown in Figure 7.

Model Training Accuracy and Loss – Character Level

This section delves into the performance metrics for Artificial Neural Networks using two types of activation functions: Rectified Linear Unit (ReLU) and Leaky Rectified Linear Unit (LReLU), specifically tailored for character recognition tasks in sign language interpretation. The model's performance is quantified using accuracy and loss metrics, which are crucial indicators of effectiveness and efficiency during the training process. Monitored these metrics over a course of 10 epochs to determine how well each model configuration performs and adapts during iterative training.

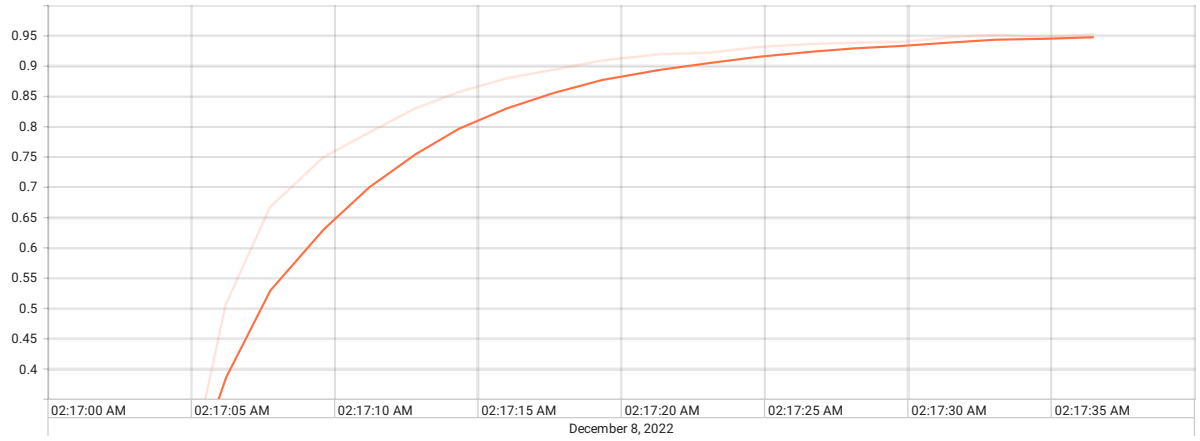


Fig 8(a). Accuracy of ReLU

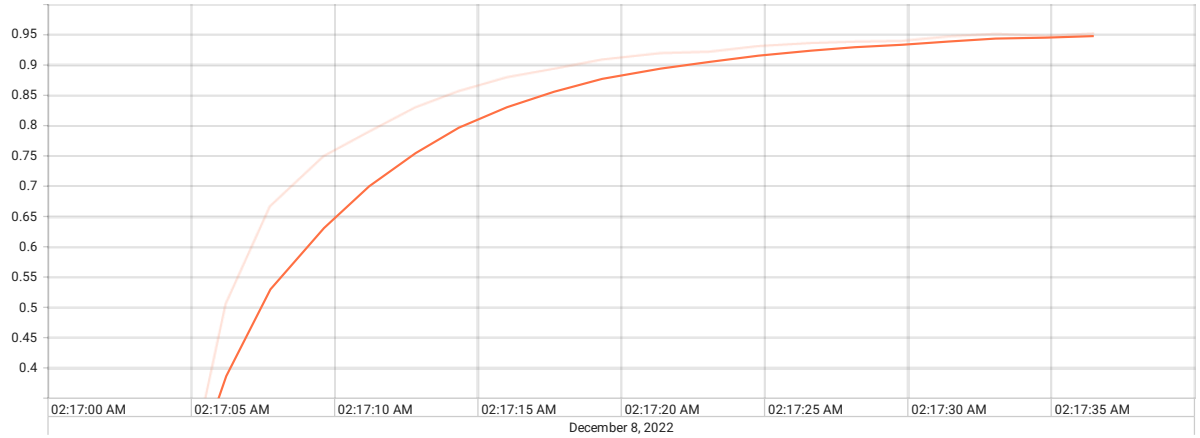


Fig 8(b). Accuracy of LReLU

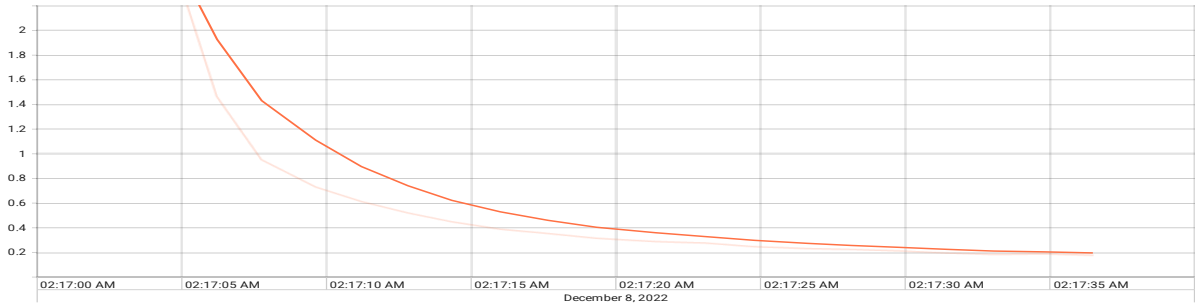


Fig 9(a). Loss of ReLU

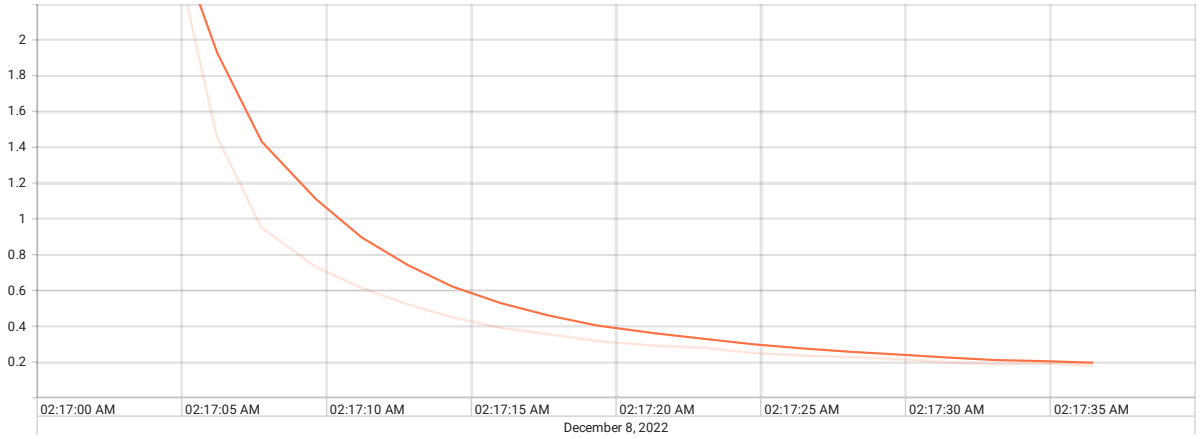


Fig 9(b). Loss of LReLU

The performance in terms of accuracy and loss does not show substantial differences when switching between the ReLU and LReLU activation functions as shown in Figure 8(a), 8(b) and Figure 9(a), 9(b). This observation indicates that, at least for the tasks and data at hand, the choice between these two activation functions does not markedly impact the outcome.

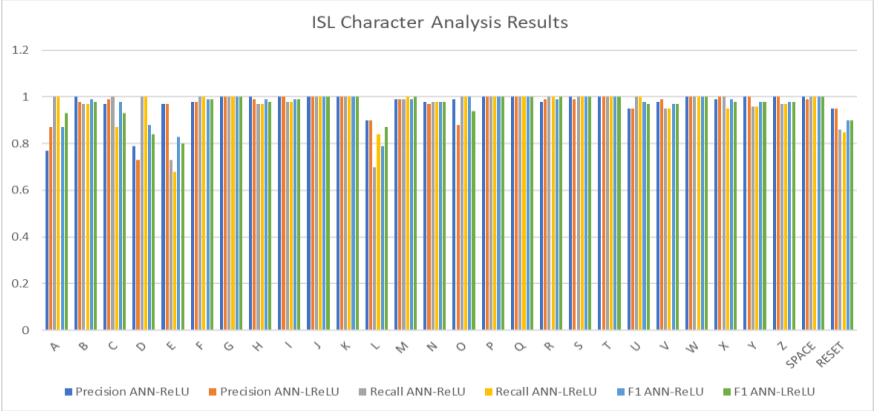
Score Analysis and Visualization

To further evaluate the models, used additional metrics and visualizations:

Table 7: Score Analysis Table - Character

Characters	Precision		Recall		F1	
	ANN-ReLU	ANN-LReLU	ANN-ReLU	ANN-LReLU	ANN-ReLU	ANN-LReLU
A	0.77	0.87	1	0.87	0.87	0.93
B	1	0.98	0.97	0.97	0.98	0.98
C	0.97	1	0.97	0.87	0.98	0.98
D	0.79	0.73	1	0.68	0.88	0.84
E	0.8	0.97	0.73	1	0.83	0.98
F	0.98	0.98	0.98	0.99	0.98	0.99
G	1	0.99	1	0.97	1	0.98
H	1	0.99	0.99	0.97	0.99	0.98
I	1	1	1	1	1	1
J	1	1	1	1	1	1
K	1	1	0.98	1	0.99	1
L	1	0.99	0.99	0.97	0.99	0.98
M	0.99	0.99	0.99	0.84	0.99	0.87
N	0.98	0.98	0.98	0.98	0.98	0.98
O	0.98	0.97	0.98	0.98	0.98	0.98
P	1	1	0.99	0.99	0.99	0.99
Q	1	1	1	1	1	1
R	0.98	0.99	1	1	0.99	0.99
S	1	1	0.99	0.99	0.99	0.99
T	1	1	0.98	0.97	0.99	0.99
U	0.95	1	0.95	0.93	0.95	0.97
V	1	1	0.97	0.99	0.98	0.99
W	1	1	0.97	0.99	0.98	0.99
X	0.99	1	1	1	1	1
Y	1	1	0.99	0.97	0.99	0.98
Z	0.99	1	0.97	0.99	0.98	0.99
SPACE	1	0.95	0.99	0.99	1	0.97
RESET	0.95	0.95	0.86	0.85	0.9	0.9

This table 7 would typically list detailed performance metrics such as precision, recall, and F1 scores for both the ReLU and LReLU models. These metrics provide a deeper understanding of each model's ability to classify correctly and balance false positives and negatives.



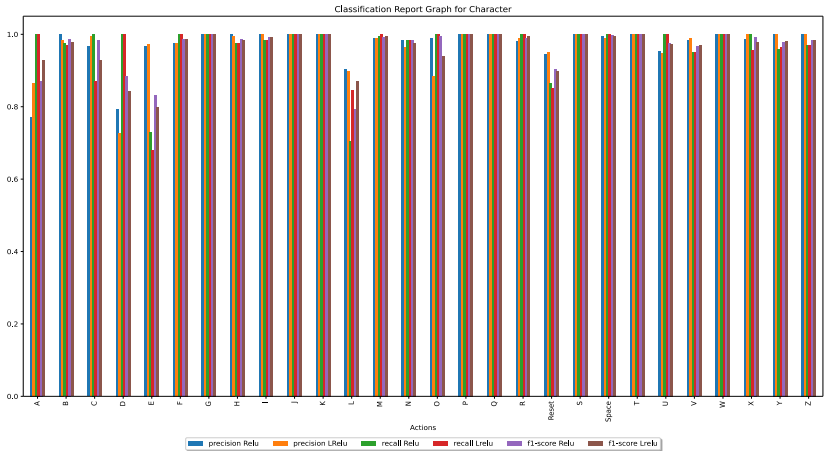


Fig 10. Score Analysis Graph – Character

This graph compares the precision, recall, and F1 scores between the two models, offering a visual representation of their comparative performance across different character recognition tasks as shown in Figure 10.

Confusion Matrices

Confusion matrices provide a straightforward way to visualize the performance of classification models by showing the actual versus predicted classifications:

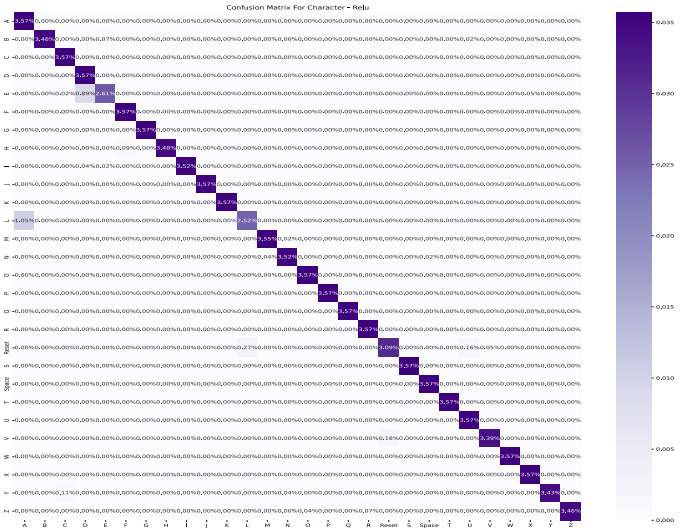


Fig 11(a): Confusion Matrix using ReLU

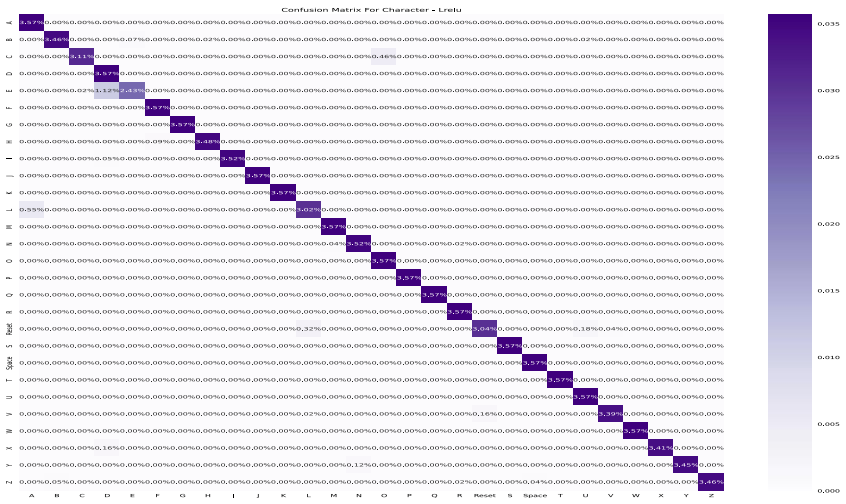


Fig 11(b): Confusion Matrix using LReLU

These visualizations and metrics collectively furnish a comprehensive overview of the models' training accuracy and loss, providing insights into the subtle effects of activation function choices on neural network performance in character-level sign language recognition as shown in Figure 11(a) and Figure 11(b).

Final Result Testing

In the final phase of study, focused on evaluating the practical application of the selected models based on the comprehensive testing and analysis conducted previously. The decision to select specific models was driven by their performance in earlier phases, aiming to optimize the accuracy and reliability of Indian Sign Language (ISL) interpretation at both the word and character levels.

Model Selection:

GRU for ISL Word-Level Interpretation: The choice of the Gated Recurrent Unit (GRU) over Long Short-Term Memory (LSTM) was influenced by GRU's demonstrated efficiency and higher performance metrics in handling word-level ISL data. GRU's ability to achieve similar or better results with less computational overhead made it the preferable option for this application.

ANN-ReLU for ISL Character-Level Interpretation: Although both ReLU and Leaky ReLU (LReLU) performed comparably well, the Artificial Neural Network (ANN) using the ReLU activation function was chosen for character-level interpretation. This decision was based on the simplicity and robustness of ReLU, which, despite the minimal performance difference, generally requires less parameter tuning and offers a slight advantage in terms of computational efficiency.

Results Visualization:

The effectiveness of these models in real-world scenarios is showcased in the figures below, which provide visual examples of the models' ISL interpretation capabilities. These examples not only illustrate the models' practical utility but also highlight their precision in interpreting complex sign language components into readable text.

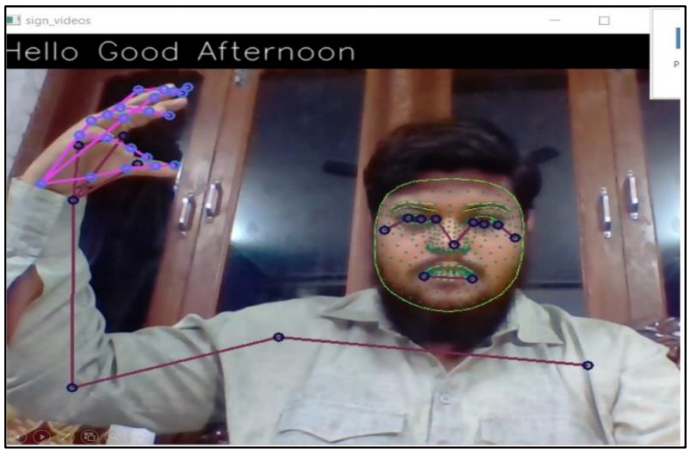


Fig 12. Final Result of ISL Word Level Interpretation

Fig 13. Final Result of ISL Character Level Interpretation

The final testing phase confirms the efficacy of the chosen models in a real-world setting, reinforcing the importance of selecting appropriate machine learning techniques for specific tasks within the domain of sign language interpretation. The examples depicted in the figure 12 and figure 13 serve as a testament to the potential of these models to significantly enhance communication accessibility for the deaf and mute community, bridging a crucial gap with effective technological solutions.

Conclusion

Sign language interpretation poses considerable challenges for real-time vision-based systems. However, as discussed in the introductory sections of this study, these challenges can be effectively addressed through various techniques and meticulous data preprocessing to ensure the accuracy of results. Crucially, providing the system with clean input data for both training and testing phases is essential for achieving the desired performance levels. The importance of proper system tuning and preprocessing to reach the expected levels of accuracy. Specifically, for vision-based continuous frame interpretation, it is necessary to employ neural networks that can maintain memory of previous data and integrate it with current frame analysis. In this context, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks are preferred because of their capability to handle long sequences of data which simple Artificial Neural Networks (ANN) may not effectively process. Experiments indicated that real-time frame interpretation is often compromised by noise, leading to incorrect results. Discovered that certain preprocessing techniques are beneficial in isolating relevant features within a frame, focusing on specific points of interest. Findings suggest that a simpler ANN, with careful hyperparameter tuning, may achieve comparable, if not superior, results under certain conditions. In some instances, the use of gloves was necessary to enhance feature recognition. Comprehensive experimental analysis reveals that for spatiotemporal data involving short sequences, GRU demonstrates superior performance compared to LSTM. For single-frame data, simple ANNs equipped with either ReLU or Leaky ReLU activation functions perform well, with no significant differences noted between these two activation types.

References

[1] Dangarwala, K. J., & Hiran, D. (2020). Deep Convolution Neural Network Model for Indian Sign Language Classification. In International Conference on Communication, Computing and Electronics Systems: Proceedings of ICCCES 2019 (pp. 35-44). Springer Singapore.

[2] Mittal, A., Kumar, P., Roy, P. P., Balasubramanian, R., & Chaudhuri, B. B. (2019). A modified LSTM model for continuous sign language recognition using leap motion. IEEE Sensors Journal, 19(16), 7056-7063.

[3] Sharma, S., Gupta, R., & Kumar, A. (2021). Continuous sign language recognition using isolated signs data and deep transfer learning. Journal of Ambient Intelligence and Humanized Computing, 1-12.

[4] Wadhawan, A., & Kumar, P. (2020). Deep learning-based sign language recognition system for static signs. Neural computing and applications, 32, 7957-7968.

[5] Fischer, S. D. (2015). Sign languages in their historical context. In The Routledge handbook of historical linguistics (pp. 442-465). Routledge.

[6] Reagan, T. (2021). Historical linguistics and the case for sign language families. Sign Language Studies, 21(4), 427-454.

[7] Ruben, R. J. (2005). Sign language: Its history and contribution to the understanding of the biological nature of language. Acta oto-laryngologica, 125(5), 464-467.

[8] Corballis, M. C. (2008). The gestural origins of language. The origins of language: Unraveling evolutionary forces, 11-23.

[9] Rajam, P. S., & Balakrishnan, G. (2011, September). Real time Indian sign language

recognition system to aid deaf-dumb people. In 2011 IEEE 13th international conference on communication technology (pp. 737-742). IEEE.

[10] Padden, C. A., & Gunsauls, D. C. (2003). How the alphabet came to be used in a sign language. Sign Language Studies, 10-33.

[11] Dour, S., & Sharma, M. M. (2015). Review of literature for the development of Indian sign language recognition system.

[12] Peguda, J., Santosh, V. S. S., Vijayalata, Y., Deepa, A., & Mounish, V. (2022, March). Speech to Sign Language Translation for Indian Languages. In 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 1131-1135). IEEE.

[13] Dutta, K. K., & GS, A. K. (2015, December). Double handed Indian Sign Language to speech and text. In 2015 Third International Conference on Image Information Processing (ICIIP) (pp. 374-377). IEEE.

[14] Goyal, S., Sharma, I., & Sharma, S. (2013). Sign language recognition system for deaf and dumb people. International Journal of Engineering Research Technology, 2(4).

[15] Dias, L., Keluskar, K., Dixit, A., Doshi, K., Mukherjee, M., & Gomes, J. (2022, July). SignEnd: An Indian Sign Language Assistant. In 2022 IEEE Region 10 Symposium (TENSYMP) (pp. 1-6). IEEE.

[16] Hore, S., Chatterjee, S., Santhi, V., Dey, N., Ashour, A. S., Balas, V. E., & Shi, F. (2017). Indian sign language recognition using optimized neural networks. In Information Technology and Intelligent Transportation Systems: Volume 2, Proceedings of the 2015 International Conference on Information Technology and Intelligent Transportation Systems ITITS 2015, held December 12-13, 2015, Xi'an China (pp. 553-563). Springer International Publishing.

[17] Athira, P. K., Sruthi, C. J., & Lijiya, A. (2022). A signer independent sign language recognition with co-articulation elimination from live videos: an Indian scenario. Journal of King Saud University-Computer and Information Sciences, 34(3), 771-781.

[18] Nair, A. V., & Bindu, V. (2013). A review on Indian sign language recognition. International journal of computer applications, 73(22).

[19] Ghotkar, A. S., & Kharate, G. K. (2014). Study of vision-based hand gesture recognition using Indian sign language. International journal on smart sensing and intelligent systems, 7(1), 96-115.

[20] Ekbote, J., & Joshi, M. (2017, March). Indian sign language recognition using ANN and SVM classifiers. In 2017 International conference on innovations in information, embedded and communication systems (ICIIECS) (pp. 1-5). IEEE.

[21] Apoorv, S., Bhowmick, S. K., & Prabha, R. S. (2020, June). Indian sign language interpreter using image processing and machine learning. In IOP Conference Series: Materials Science and Engineering (Vol. 872, No. 1, p. 012026). IOP sPublishing.

[22] Kulkarni, A., Kariyal, A. V., Dhanush, V., & Singh, P. N. (2021, September). Speech to indian sign language translator. In 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021) (pp. 278-285). Atlantis Press.

[23] Adithya, V., Vinod, P. R., & Gopalakrishnan, U. (2013, April). Artificial neural network-based method for Indian sign language recognition. In 2013 IEEE conference on information & communication technologies (pp. 1080-1085). Ieee.

[24] Aloysius, N., & Geetha, M. (2020). Understanding vision-based continuous sign language recognition. Multimedia Tools and Applications, 79(31), 22177-22209.

[25] Ananthanarayana, T., Srivastava, P., Chintha, A., Santha, A., Landy, B., Panaro, J., ... & Nwogu, I. (2021). Deep learning methods for sign language translation. ACM Transactions on Accessible Computing (TACCESS), 14(4), 1-30.

[26] Lee, B. G., Chong, T. W., & Chung, W. Y. (2020). Sensor fusion of motion-based sign language interpretation with deep learning. Sensors, 20(21), 6256.

[27] Ru, J. T. S., & Sebastian, P. (2023, May). Real-Time American Sign Language (ASL) Interpretation. In 2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN) (pp. 1-6). IEEE.

[28] Cui, R., Liu, H., & Zhang, C. (2017). Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7361-7369).

[29] Alharthi, N. M., & Alzahrani, S. M. (2023). Vision Transformers and Transfer Learning Approaches for Arabic Sign Language Recognition. Applied Sciences, 13(21), 11625.